Proiect

Elemente avansate de programare

- Curăţătorie haine –

1. Tema proiectului

O firmă care se ocupă de curăţarea hainelor are nevoie de o aplicaţie care să monitorizeze intrările clienţilor şi să le stocheze în format digital pentru o mai bună gestionare şi pentru a putea obţine rapoarte de activitate mai rapid.

1. Echipa

* Costache Costin - grupa 253
* Nicolescu Bristena – grpa 253

1. Etape de dezvoltare
2. Dezvoltarea bazei de date
3. Documentarea în privinţa diverselor framework-uri şi librării disponibile
4. Am decis că aplicaţia va fi web şi am ales ca punct de plecare librăria spark - basic-web-app example : <https://github.com/tipsy/spark-basic-structure>
5. Verificat Login
6. Adăugare servicii în baza de date
7. Adăugare bon ca user / admin
8. Listare bonuri
9. Editare statusuri produse / bonuri
10. Rapoarte
11. Administrare utilizatori
12. Descrierea aplicaţiei

Aplicaţia are 2 tipuri de useri :

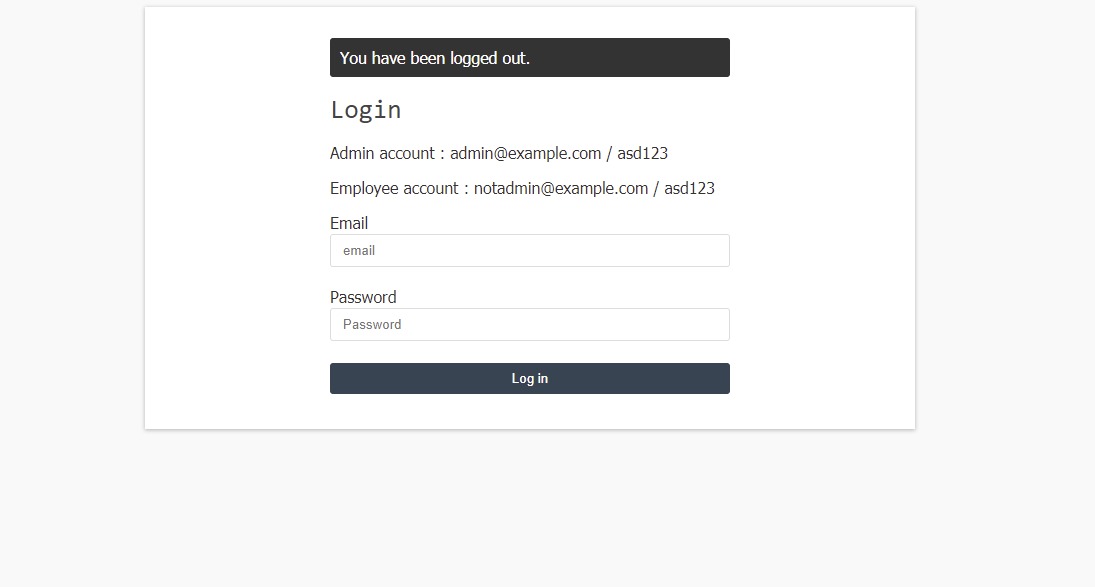
* **Angajatul** se ocupă de
* introducerea hainelor pe un bon atunci când vine un client
* gestionarea hainelor (schimbarea statusului din murdare în curatate)
* schimbarea statusului bonului în “returnat” (atunci când clientul îşi primeste hainele curăţate)
* **Managerul** poate să
* gestioneze conturile angajaţilor (să creeze conturi noi sau să le modifice pe cele existente)
* să vadă raportul încasărilor din luna curentă (cu procentaj pentru fiecare serviciu)
* să acceseze istoricul comenzilor unui client de-a lungul timpului dupa email-ul acestuia

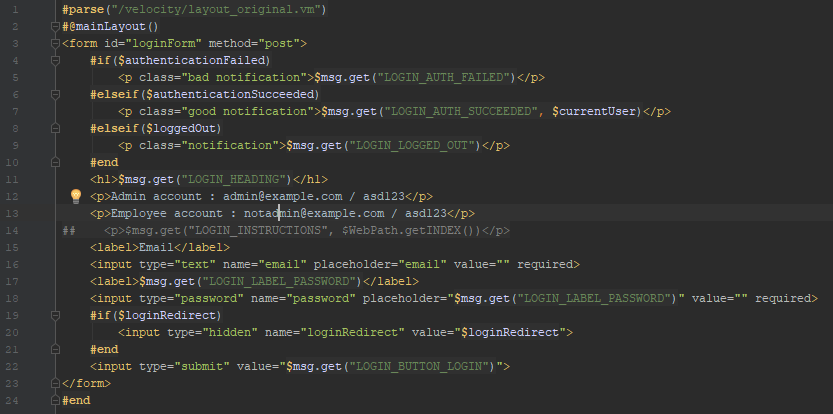
1. Persistenţa datelor

Datele sunt stocate într-o bază de date MySQL folosind hybernate pentru a mapa o instanţă din baza de date într-un obiect java.

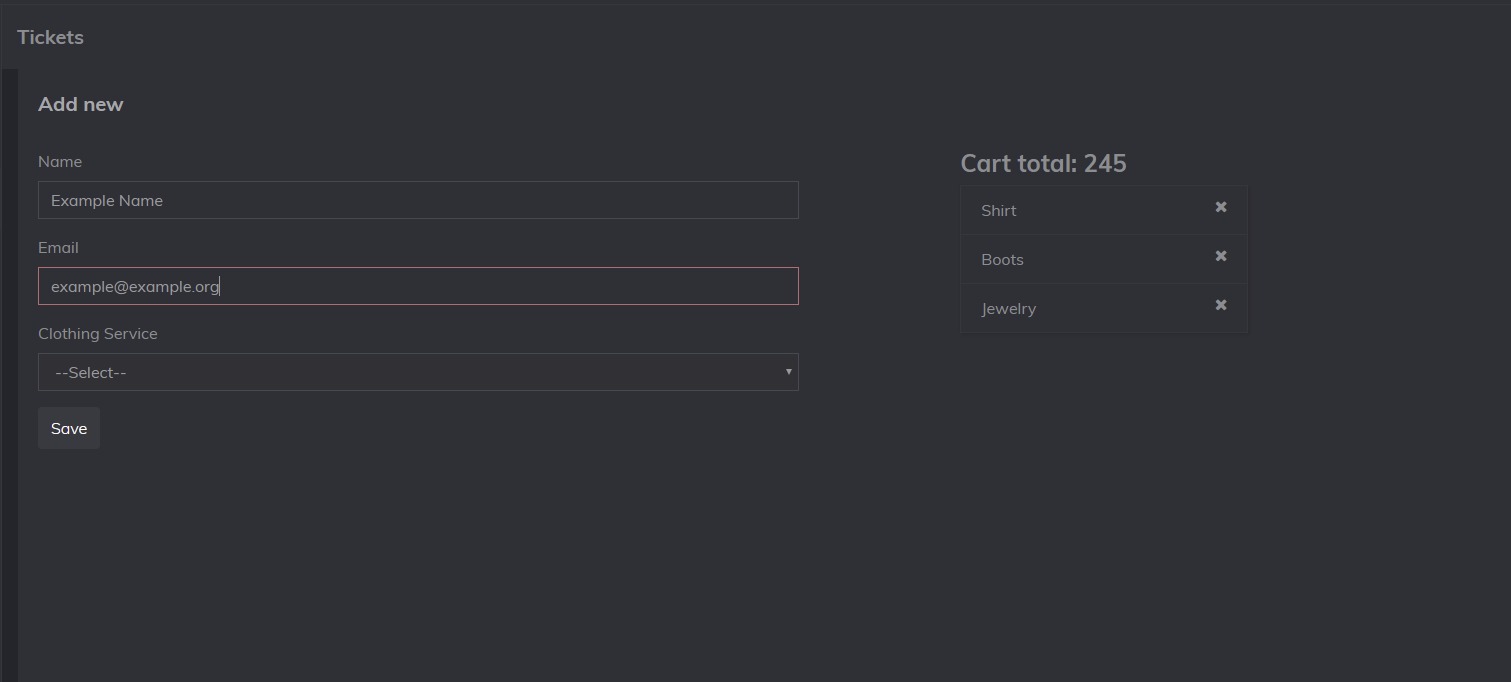
1. Secţiuni ale aplicaţiei

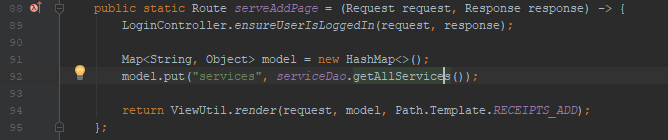
* Login

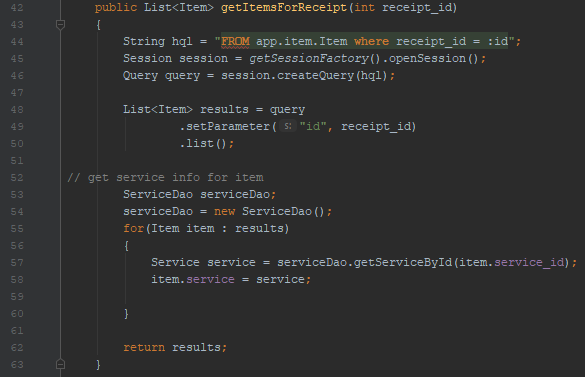




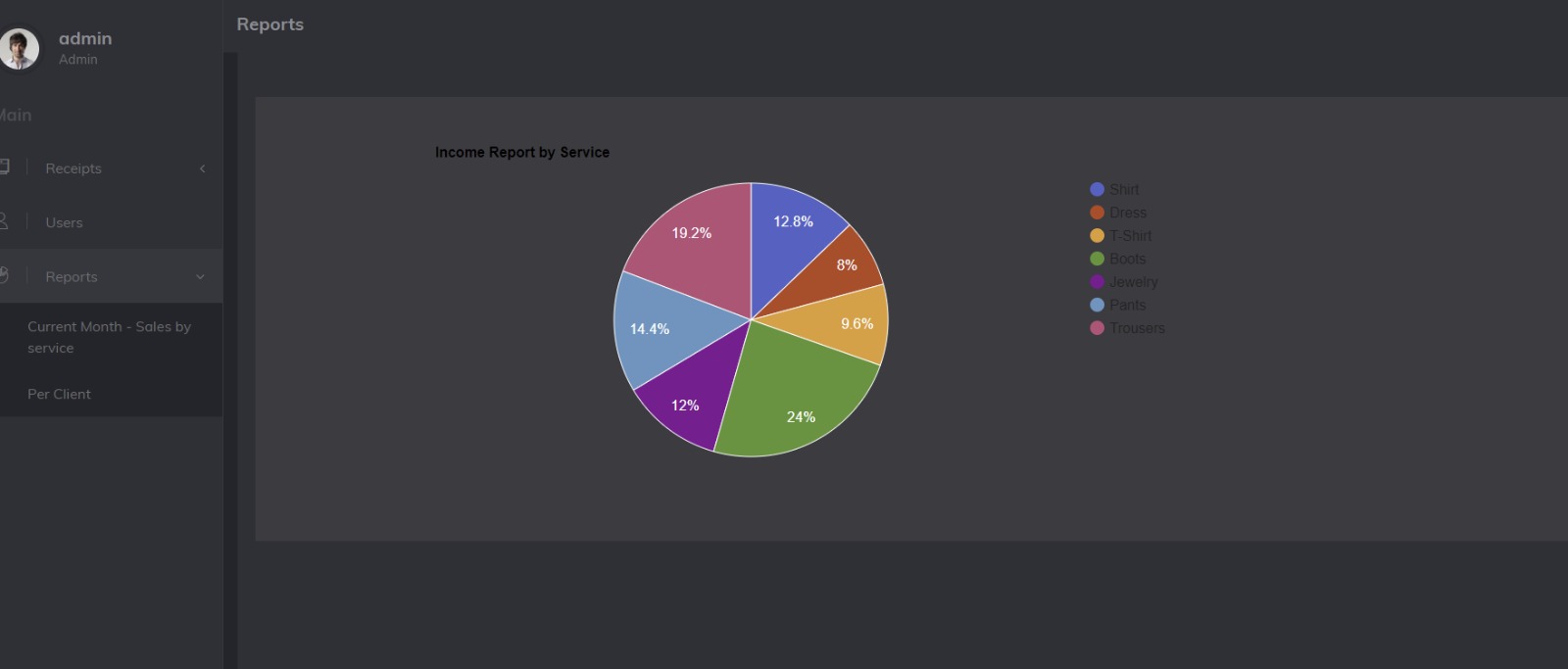
* Adăugarea unui nou bon

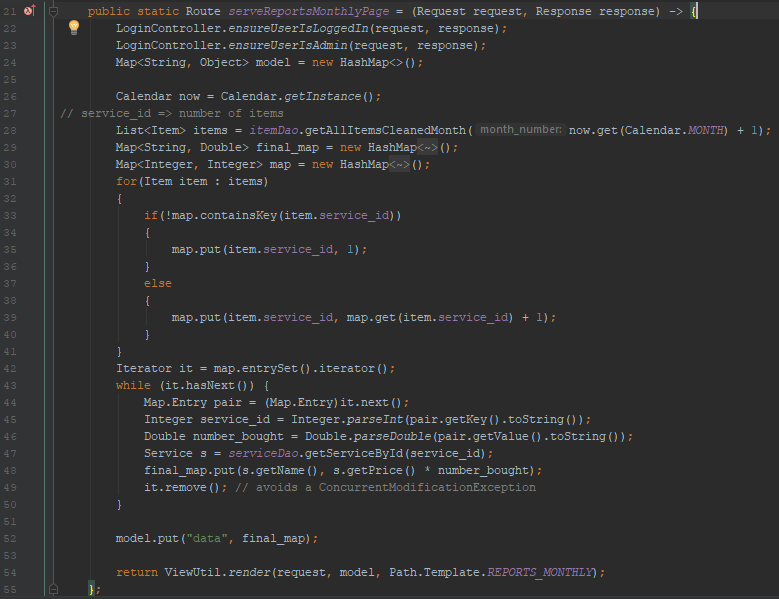






* Raport lunar





1. Ce a mers / nu a mers bine? Ce am învăţat din dezvoltarea acestui proiect?

A mers bine:

1. sistemul de login funcţional datorită Spark framework
2. salvarea sesiunilor datorită Spark framework

Nu a mers bine :

* 1. bcryptul implicit cu care venea Spark => am instalat google-guava
  2. să facem update + persist în baza de date a unei entităţi fără să folosim tranzacţii

Am învăţat că :

* 1. este mult mai uşor să trimitem date în formă de obiect pe care să le formatez în json în view decat să formatăm un string pe care doar să îl pasăm în view (de ajutor în special la parsarea datelor pentru librăriile de grafice) => nu trebuie trimise datele de 2 ori, odată ca array de obiecte şi odată ca json encodat şi formatat
  2. este foarte complicat să includem o clasă Helper care să fie disponibilă în TOATE template-urile de velocity pentru a ne stoca nişte constante / a avea funcţii ajutatoare
  3. trebuie pornit hibernate auto-commit , alftel nimic nu persistă în BD fără să dăm flush la sesiune